

This Page Is Inserted by IFW Operations
and is not a part of the Official Record

BEST AVAILABLE IMAGES

Defective images within this document are accurate representations of the original documents submitted by the applicant.

Defects in the images may include (but are not limited to):

- BLACK BORDERS
- TEXT CUT OFF AT TOP, BOTTOM OR SIDES
- FADED TEXT
- ILLEGIBLE TEXT
- SKEWED/SLANTED IMAGES
- COLORED PHOTOS
- BLACK OR VERY BLACK AND WHITE DARK PHOTOS
- GRAY SCALE DOCUMENTS

IMAGES ARE BEST AVAILABLE COPY.

As rescanning documents *will not* correct images,
please do not report the images to the
Image Problem Mailbox.



(19)

Europäisches Patentamt

European Patent Office

Office européen des brevets



(11)

EP 0 961 193 A2

(12)

EUROPEAN PATENT APPLICATION

(43) Date of publication:
01.12.1999 Bulletin 1999/48(51) Int. Cl.⁶: G06F 1/00, G06F 12/14

(21) Application number: 99201705.3

(22) Date of filing: 28.05.1999

(84) Designated Contracting States:
 AT BE CH CY DE DK ES FI FR GB GR IE IT LI LU
 MC NL PT SE
 Designated Extension States:
 AL LT LV MK RO SI

(30) Priority: 29.05.1998 US 87229
 29.05.1998 US 87195
 29.05.1998 US 87262
 29.05.1998 US 87230 P

(71) Applicant:
 Texas Instruments Incorporated
 Dallas, Texas 75251 (US)

(72) Inventor: Laczko Sr., Frank L.
 Allen, Collin County, Texas (US)

(74) Representative: Holt, Michael
 Texas Instruments Limited,
 P.O. Box 5069
 Northampton NN4 7ZE (GB)

(54) Secure computing device

(57) A secure computing system (100) stores a program, preferably the real time operating system (210), that is encrypted with a private key. A boot ROM (135) on the same integrated circuit as the data processor and inaccessible from outside includes an initialization program and a public key corresponding to the private key. On initialization the boot ROM decrypts at least a verification portion of the program. On verification normal operation is enabled. On non-verification, the system could be disabled, or that application program could be disabled. A diagnostic program is stored at predetermined non-relocatable physical address in memory. The program is made non-relocatable using a special table look-aside buffer (137) having a fixed virtual address register (611) and a corresponding fixed physical address register (641). The secure computing system

prevents unauthorized use of compressed video data stored in a first-in-first-out memory buffer by encrypting the compressed video data stream using at least a part of the chip identity number as an encryption key (703). The data is recalled from memory (705) and decrypted (706) as needed for video decompression. The debugger/emulator tool commonly employed in program development is protected by a private encryption key used to encrypt at least verification token for the program. Upon each initialization of the debugger/ emulator, the secure computer system decrypts the verification token employing public decryption key (805) to indicate whether the program is secure or non-secure.

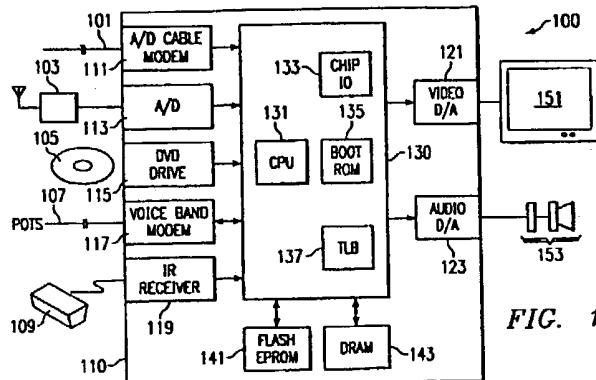


FIG. 1

Description**TECHNICAL FIELD OF THE INVENTION**

[0001] The technical field of this invention is secure computing systems, especially computer systems that may execute after manufacture field provided programs secured to prevent the user from unauthorized use of selected computer services. The computer system may also be functionally reprogrammable in a secure manner.

BACKGROUND OF THE INVENTION

[0002] There are currently many methods to deliver video programming to users of television besides over the air broadcast. Numerous service providers are available to supply this programming to television viewers. Most of these service providers vend a hierarchy of services. Typically there is a basic service for a basic fee and additional services available for an additional fee. The basic services typically include the broadcast network programming, cable superstations, music and sports programming. These basic services are typically supported by advertising. These basic programming services thus operate on the same economics as over the air broadcast television. The additional services typically include the so called "premium" programming such as sports and movies. These premium programming services are typically not advertiser supported. These are perceived by the television user as higher value services and television users are willing to pay their service providers additional fees for these services. The service provider passes much of this additional fee to the content providers as their compensation for supplying the programming. There may be one or several tiers of these premium services made available by the service providers. At the top of this programming hierarchy is pay per view programming. Pay per view programming typically includes music concerts and sporting events perceived as time sensitive and highly valuable by the television users. Pay per view may also include video on demand, where the television user requests a particular movie be supplied. This hierarchy of service exists for all current alternative methods of program delivery including television cable, over the air microwave broadcast and direct satellite television.

[0003] Reception of such alternative programming services has required an additional hardware appliance beyond the user provided television receiver since the beginning of cable television. Initially this additional hardware appliance merely translated the frequency of the signal from the transmission frequency to a standard frequency used in broadcast television. Such a standard frequency is receivable by the user provided television receiver. This additional hardware appliance is commonly known as a "set top box" in reference to its typical deployment on top of the television receiver. Cur-

rent set top boxes handle the hierarchy of security previously described.

[0004] In the past these set top boxes have been fixed function machines. This means that the operational capabilities of the set top boxes were fixed upon manufacture and not subject to change once installed. A person intending to compromise the security of such a set top box would need substantial resources to reverse engineer the security protocol. Accordingly, such fixed function set top boxes are considered secure. The future proposals for set top boxes places the security assumption in jeopardy. The set top box currently envisioned for the future would be a more capable machine. These set top boxes are expected to enable plural home entertainment options such as the prior known video programming options, viewing video programming stored on fixed media such as DVD disks, Internet browsing via a telephone or cable modem and playing video games downloaded via the modem or via a video data stream. Enabling the set top box to be programmed after installation greatly complicates security. It would be useful in the art to have a secure way to enable field reprogramming of set top boxes without compromising the hierarchy of video programming security.

SUMMARY OF THE INVENTION

[0005] The present application discloses a secure computing system. A program, preferably the secure computing system real time operating system, is encrypted with a private key. The data processor includes a boot ROM on the same integrated circuit that is inaccessible from outside the integrated circuit. The boot ROM includes the public key corresponding to the private key used to encrypt the program. On initialization the boot ROM decrypts at least a verification portion of the program. This enables verification or non-verification of the security of the program. The boot ROM may store additional public keys for verification of application programs following verification of the real time operating system. Alternatively, these additional public keys may be stored in the non-volatile memory.

[0006] On verification of the security of the program, normal operation is enabled. There are several remedial actions that can take place on non-verification. The system could be disabled, or in the case of non-verification of an application following verification of the real time operating system only that application program could be disabled. The system could notify the system vendor of the security violation using the modem of the secure computing system.

[0007] A diagnostic program can check the security of a program. The program is stored at predetermined physical address in memory. Relocation of these physical addresses where the program is stored is prevented. The diagnostic program is loaded and checks the program at the predetermined physical address against a standard. The diagnostic program then indicates that

the program is verified as secure if it meets the standard or non-verified as secure if it does not meet the standard.

[0008] The program is made non-relocatable using a special table look-aside buffer. The table look-aside buffer has a fixed virtual address register and a plurality of writable virtual address registers. Each of these virtual address registers has a comparator and a corresponding physical address register. The physical address register corresponding to the fixed virtual address register is also fixed. The fixed virtual address register and the fixed physical address register encompass the range of addresses where the program is stored. The fixed virtual address register and the fixed physical address register are preferably mask programmable in manufacture via a metal layer

[0009] The fixed virtual address register and the fixed physical address register may be registers ostensibly writable via the instruction set architecture. In this case, attempts to write to these registers do not change their contents. In addition, it is preferable that attempts to write to these registers produce no faults or exceptions. Alternatively, the fixed virtual address register and the fixed physical address register may not be accessible via the instruction set architecture.

[0010] The disclosed embodiment of the secure computing system prevents unauthorized use of compressed video data stored in a first-in-first-out memory buffer in a set top box. Current video compression techniques do not compress data uniformly. For this reason a uniform compressed video data rate does not translate into a uniform decompressed video data rate. Typical set top boxes employ off chip DRAM as a first-in-first-out (FIFO) buffer to prevent the decompression process from overflowing or underflowing. The memory bus traffic between the data processor and the portion of memory used as the FIFO buffer is subject to interception and unauthorized use.

[0011] The data processor is disposed on a single integrated circuit. This data processor includes a chip identity read only register storing a unique chip identity number. This unique chip identity number is fixed during manufacture by, for example, laser probing or selective activation of fuse or antifuse links in the chip identity register. The data processor encrypts the compressed video data stream using at least a part of the chip identity number as an encryption key. This encrypted data is stored in the memory area serving as the FIFO buffer. The data is recalled from memory as needed for video decompression. The date processor then decrypts the recalled data employing at least a part of the chip identity number as the decryption key.

[0012] Using this technique the compressed video data stream temporarily stored in compressed form in the FIFO buffer can only be read by the particular data processor having the unique chip identity number. Since the chip identity number is unique to that particular data processor the video data cannot be processed by

another data processor, even another identical set top box system without breaking the code. The encryption and decryption is transparent to the user requiring only a small additional processing capacity within the data processor.

[0013] Another aspect of this invention concerns the security of a computer system when used with a debugger/emulator tool commonly employed in program development. Without special procedures to limit the operation of the debugger/emulator tool, the security of the computer system would be subject to compromise.

[0014] The disclosed embodiment of the secure computing system uses an encryption system employing a private encryption key and a public decryption key. The private encryption key is used to encrypt at least a verification token for the program. The public decryption key corresponding to the private encryption key is stored at the secure computing system. Upon each initialization of the debugger/emulator for the secure computing system a security screen is performed. This involved determining if the program is secure program or a non-secure program. The secure computer system decrypts the verification token employing public decryption key. This decrypted verification token indicates the program as a secure program or a non-secure program.

If the program is a secure program, then the debugger/ emulator is operated in a process mode. The process mode permits the debugger/emulator access to the program while prohibiting access to at least one security feature of the secure computing system. If the program is a non-secure program, then the debugger/emulator is operated in a raw mode. The raw mode permits the debugger/emulator to access all features of the secure computing system.

[0015] A further security layer is used for operating system development intended for the secure computing system. Each data processor includes a unique chip identity number stored in a read only chip identity register. If the program is a secure program, then the debugger/emulator reads the chip identity number. A certain subset of the chip identity numbers and only this subset will permit the debugger/emulator to operate in the raw for a secure program. If the chip identity number does not fall within this subset, then the debugger/emulator can only operate in the process mode.

BRIEF DESCRIPTION OF THE DRAWINGS

[0016] The present invention will now be further described by way of example, with reference to the accompanying drawings in which:

Figure 1 is a block diagram of one embodiment of the disclosed secure computing system;

Figure 2 is an example memory map of the boot read only memory of the digital media processor illustrated in Figure 1;

Figure 3 is an example memory map of the non-vol-

atile memory of the set top box illustrated in Figure 1;

Figure 4 is an example memory map of the read write memory illustrated in Figure 1;

Figure 5 is a flow chart of the initial operation including the operating system verification of the digital media processor illustrated in Figure 1;

Figure 6 is a flow chart of the process for verification of an application to the set top box illustrated in Figure 1;

Figure 7 is a flow chart of the process of verification of a downloaded application program;

Figure 8 is a schematic diagram of a translation look aside buffer preventing virtual memory relocation of a certain page of memory of the digital media processor of Figure 1;

Figure 9 is a flow chart of the process of encrypting and decrypting compressed video data temporarily stored in random access memory; and

Figure 10 is a flow chart of the process of mode selection in a hardware debugger/emulator.

DETAILED DESCRIPTION OF PREFERRED EMBODIMENTS

[0017] The set top box of the future will enable home entertainment options such as the known video programming options, viewing video programming stored on fixed media such as digital video disks (DVD), Internet browsing via a telephone or cable modem and playing video games downloaded via the modem or via a video data stream. Such a variety of capability can only be provided by a fully programmable data processor which can receive and run downloaded programs. This opens up a host of security issues. Since much of the utility of the system depends on being able to download various applications, the possibility also exists for an unauthorized application being downloaded. Such an unauthorized application may be deliberately written to compromise the hierarchy of security.

[0018] Fully programmable set top boxes are vulnerable to three main types of attacks. An unauthorized application may interact with the operating system, possibly bypassing security. The set top box non-volatile memory may be replaced with modified resident applications, but with the original operating system. The non-volatile memory may be replaced with a new operating system. The most important item to protect is the operating system. If the operating system is compromised, an unauthorized person can do almost anything, including disguising the fact that the operating system is compromised.

[0019] Figure 1 illustrates in schematic form the parts of a versatile, programmable set top box system 100. Set top box system 100 is responsive to inputs from: television cable 101; direct satellite receiver front end 103, digital video disk (DVD) 105; an ordinary telephone line 107; and infrared remote control 109. These inputs are

conventional and need not be more fully described here. Any interaction of these conventional inputs with the parts of the disclosed embodiment of the secure computing system will be more fully described below.

[0020] The central part of set top box system 100 is the set top box 110. Set top box 110 includes various interfaces for the inputs including: video analog-to-digital converter 111 connected to the television cable 101, which may optionally include a cable modem; video analog-to-digital converter 113 connected to direct satellite receiver front end 103; a DVD driver 115 capable of receiving and reading DVD 105; voice band modem 117 connected to telephone line 107; and infrared receiver 119 capable of receiving the infrared signals from infrared remote control 109.

[0021] Set top box 110 includes several output devices coupled to digital media processor 130. Video digital-to-analog converter 121 receives a video data stream from digital media processor 130 and supplies an corresponding video signal to television receiver 151. Typically the desired video data stream is modulated upon a carrier having a frequency which the television receiver 151 can normally receive. It is contemplated that video media processor 130 in cooperation with video digital-to-analog converter 121 will be capable of producing a video signal in a plurality of formats. Upon set up of set top box system 100 the particular format will be selected to correspond to the capability of the particular television receiver 151 employed. Audio digital-to-analog converter 123 receives an audio data stream from digital media processor 130 and supplies a base band audio signal to audio system 153. It is contemplated that this audio signal may encompass plural audio channels (i.e. left and right channels for stereo). It is also contemplated that any particular video source may include plural encoded audio data streams such as alternative languages, descriptive video or other separate audio programs (SAP). Note also that the audio data stream will typically also be modulated on the same carrier as the video signal for reception and demodulation by television receiver 151.

[0022] The intelligent part of set top box 110 is digital media processor 130. Digital media processor 130 is preferably embodied in a single integrated circuit. Note that in order for set top box 110 to be fully secure as intended, central processing unit 131 and boot ROM 135 must be located on the same integrated circuit. Digital media processor 130 includes central processing unit 131. Central processing unit 131 is illustrated generically and is not intended to limit the structure employed. Central processing unit preferably includes data processing capability for control functions required for selection of operating mode, channel tuning, security functions and the like. Central processing unit preferably also includes digital signal processing capability for decompressing compressed video and audio signals, decrypting encrypted video signals, converting the

received video to the format of the user's television receiver, operating as a "software" cable modem and voice band modem and demodulating the signal from infrared remote control 109. Central processing unit 131 may include a microprocessor and a digital signal processor, a single data processor capable of all the necessary functions or a multiprocessor. The exact nature of central processing unit, except for details noted below, is not relevant to disclosure of the present application.

[0023] Digital media processor 130 further includes chip identity register 133. Chip identity register 133 is a programmable readable register holding an identity number unique to the integrated circuit embodying digital media processor 130. This identity number is preferably implemented as taught in U.S. Patent Application No. 08/813,887 entitled *Circuits, Systems, and Methods for Uniquely Identifying a Microprocessor at the Instruction Set Level* and filed March 7, 1997. As described in this patent application, the unique identification code is formed in a read-only data register by laser probing following integrated circuit test. The unique chip identity number may be specified via selective blowing of fuse or antifuse links or other techniques. This identity number permits a program to verify the exact identity of the particular digital media processor 130 used in the set top box 110.

[0024] Digital media processor 130 includes boot read only memory (ROM) 135. Digital media processor 130 is constructed so that central processing unit 131 begins executing program instructions stored within boot ROM upon each initial application of electric power. An exemplary memory map of boot ROM 135 is illustrated in Figure 2. Those skilled in the art will realize that the exact order of storage of the various parts is not as important as the existence of the detailed data types. Boot ROM 135 includes self boot code 201. Self boot code 201 is the program instructions initially executed by central processing unit 131 upon each initial application of electric power to digital media processor 130. In addition the known processes for initializing computer systems, self boot code 201 also includes verification program code 202. Verification program code 202 will be further described below in conjunction with Figure 5. Boot ROM 135 also includes a public signature keys. These public signature keys include real time operating system (RTOS) public signature key 203, first application public signature key 205, second application public signature key, to the Nth application public signature key 207. These public signature keys are employed in verification of the authorization of programs in a manner that will be further described below.

[0025] Digital media processor 130 also includes table look-aside buffer (TLB) 137. Table look-aside buffer 137 is employed to enhance security during virtual memory operation in a manner further described below.

[0026] Set top box 110 includes flash (electrically programmable read only memory) EPROM 141 bi-directionally coupled to digital media processor 130. Flash

5 EPROM 141 serves as the non-volatile memory for set top box system 100. This is known as a non-volatile memory because it retains its contents when electric power is turned "OFF". Non-volatile memory is needed for the real time operating system (RTOS) and for resident applications. Figure 3 illustrates an exemplary memory map of flash EPROM 141. Flash EPROM 141 includes the real time operating system (RTOS) 210. RTOS 210 includes program code enabling digital media processor 130 to receive and process various data streams as they are received, i.e. in "real" time. RTOS 210 also enables digital media processor 130 to respond to operator control via infrared remote control 109 and infrared receiver 119. RTOS 210 includes a signature portion 211 whose use will be further described below. Flash EPROM 141 also includes program code for the first resident application 220 with its corresponding signature portion 221. Likewise, flash EPROM 141 includes program code for the second resident application 230 and its corresponding signature portion 231 and program code for other resident applications to the Mth resident application 240 and its corresponding signature portion 241. Flash EPROM 141 optionally includes additional public keys including N + 1st public key 251, N + 2nd public key 253 to N + Pth public key 255. These additional public signature keys are similar to the N public signature keys stored in boot ROM 135. Their use will be detailed below.

10 [0027] Set top box 110 further includes dynamic random access memory (DRAM) 143 bi-directionally coupled to digital media processor 130. DRAM 143 is a volatile memory that serves as read/write memory to temporarily store transient data during normal operations. DRAM 143 is preferably embodied by synchronous memory employing a RAMBUS interface. Figure 4 illustrates an exemplary memory map of DRAM 143. DRAM 143 stores the memory resident part 261 of the real time operating system. Depending upon the particular status of set top box system 100 this memory resident part 261 of the RTOS may differ as known in the art. DRAM 143 stores the memory resident parts 263 of the currently running application or applications. These applications may be resident applications stored in flash EPROM 141 or transient applications stored in other parts of DRAM 143. Depending upon the status of set top box system 100, there may be various applications running and their immediately accessible parts will be stored in DRAM 143 for faster access than from flash EPROM 141. DRAM 143 also stores transient data 265. DRAM 143 stores the program code of various transient applications such as first transient application 271, second transient application 273 to Qth transient application 275. Transient applications are those loaded via cable modem 111, voice band modem 117 or DVD drive 115 that are

intended for use only during the current session of set top box system 100. These may include video games, Internet browsing and the like. These transient applications are loaded into DRAM 143 each time they are used and then discarded. DRAM 143 also stores compressed video in a first-in-first-out (FIFO) buffer 280. Video data from television cable 101, direct satellite receiver front end 103 and DVD 105 will generally be transmitted in compressed form. This saves transmission bandwidth and storage space. One of the tasks of digital media processor 130 is to decompress the video data. Current video compression formats (such as MPEG2) and all contemplated future video compression formats are non-linear. That is, the different portions of the video data stream are compressed to differing degrees. Thus a constant rate of received video data represents varying amounts of video. Following decompression, digital media processor must supply video data in a constant rate to be viewed. Compressed video FIFO buffer 270 is necessary to smooth out the variations in the rate of receipt. This permits the decompression process to neither overflow with too much compressed data nor underflow with no compressed data ready for decompression. This is possible because the compressed video data stream represents a constant rate video data stream that is to be viewed. Thus the overall average compressed video data rate corresponds to the constant real time viewing rate.

[0028] Figure 5 is a flow chart 300 of an example of digital media processor 130 operations controlled by boot ROM 135. Upon each initial application of electric power to set top box system 100, digital media processor begins executing the program stored in a predetermined location within boot ROM 135. Those portions of this program within boot ROM 135 relevant to disclosure of the present application are illustrated in Figure 5. Program 300 first initialized digital media processor 130 (processing block 301). This process would include clearing registers and caches, setting the initial operating mode and the like, in a manner known in the art. Following initialization of the processor, program 300 reads the signature portion 211 of RTOS 210 stored in flash EPROM 141 (processing block 302). Program 300 next reads the RTOS public key 203 from boot ROM 135 (processing block 303). Next program 300 verifies the signature portion 211 of RTOS 210 (processing block 304). In accordance with the known art of public key encryption such as the RSA algorithm, signature portion 211 is produced by operating upon all of RTOS 210 with a secret private signature key. The original data of signature portion 211 is recovered by a reverse process employing RTOS public signature key 203 stored in boot ROM 135. This signature verification process takes into account what is known as a "trap door" function. It is a very difficult process to produce a particular signature portion knowing only the public key. A change of any portion of RTOS 210 is very likely to result in a change in the signature portion 211 in a manner that cannot be

predicted from only the RTOS public signature key 203. Thus it is possible to detect any change in RTOS 210 employing the signature portion 211.

[0029] Following the verification, program 300 tests the verified signature portion to determine if RTOS 210 supports secure applications (decision block 305). The preferred embodiment of the secure computing system of the present application contemplates that digital media processor 130 could be embodied in applications not requiring the security of set top boxes. In such applications, the verified signature portion 211 indicates that the RTOS need not be secured. Note that even a non-secure RTOS must have its stub verified. Failure of the signature verification is fatal whether the RTOS is secure or non-secure. Program 300 bypasses other steps and starts RTOS 210 (processing block 310) if this signature portion 211 indicates a non-secure use. This will typically involve loading at least a portion of RTOS 210 into DRAM 143. It is anticipated that DRAM 143 will allow much faster memory access than flash EPROM 141. Thus loading portions of RTOS 210 into DRAM 143 will enable faster operation.

[0030] If the verified signature portion indicates that RTOS 210 is to support secure applications (decision block 305), then program 300 tests to determine if RTOS 210 can be verified as correct (decision block 306). As described above, the trap door function of the private key signature with public key signature makes it a very difficult task to modify RTOS 210 without producing an unpredictable modification of signature portion 211. Thus the initial program stored in boot ROM 135 will almost certainly be able to detect unauthorized modification of RTOS 210. This verification of RTOS 210 permits the vendor of set top box system 100 to be confident of the security of the system.

[0031] If the verified signature portion is not verified as secure, then program 300 indicates that RTOS 210 is non-secure (processing block 307). Thereafter program 300 takes remedial action (processing block 308). This remedial action can take many forms. At the most severe, this remedial action could be complete disablement of set top box 110. Shutting down media processor 130 will disable set top box 110 since it is the intelligence of set top box 110. In most secure applications running a non-verified RTOS would be considered very dangerous and the only reasonable remedial action is disabling set top box 110. In a few cases a less severe remedial action may be appropriate. As a less severe remedial measure, digital media processor 130 could be programmed to no longer interact with video data streams from television cable 101, direct satellite receiver front end 103 and/or DVD 105. This mode may permit running local only transient applications. The remedial action could include signaling the set top box vendor or service provider of the security violation via cable modem 111 or voice band modem 117. The recipient of this notification could then determine either automatically or manually how to deal with the security

violation. One method of responding to such a notification of a security violation is to download via cable mode 111 or voice band modem 117 an authorized copy of the RTOS for storage in flash EPROM 143, overwriting the unauthorized copy. Another method is to download a diagnostic program which will verify and determine the extent of the security violation. At the least severe level most suitable for service providers who supply only advertiser supported program material, is to ignore the security violation and permit operation of the non-secure RTOS.

[0032] If the verified signature portion is verified as secure, then program 300 indicates that RTOS 210 as verified (processing block 309). Thereafter program 300 starts operation of RTOS 210 (processing block 310). As described above this would typically involve copying at least portions of RTOS 210 from flash EPROM 141 to DRAM 143. Following such a copying, program control would be transferred to the RTOS copy in DRAM 143 via a jump instruction. RTOS 210 then enables all the authorized functions of set top box system 100.

[0033] The entire RTOS could be encrypted using the private key as an alternative to employing merely a signature verification process. The steps illustrated in Figure 5 would be similar except that the entire RTOS must be decrypted using the public key rather than just the signature portion. In this event, the decrypted RTOS would be copied to a operating portion of DRAM 143 upon verification. Thereafter program control would be passed to this copy of the RTOS from the boot ROM program via a jump instruction. In this case a non-verified RTOS even if copied into the same part of DRAM 143 will not operate. An incorrect decryption of an unauthorized RTOS 210 would likely result in an inoperable operating system. Thus the remedial action is this case disables set top box 110. Note the use of a private key to encrypt and a public key to decrypt is the reverse of the usual private key/public key system. Currently, only the RSA system is known to permit this reverse use.

[0034] Figure 6 is a flow chart 400 of an example of digital media processor 130 operations when called to load and run a resident application. Following the command to start the resident application program (processing block 401), program 400 reads the corresponding signature portion of the resident application stored in flash EPROM 141 (processing block 402). Program 400 next reads the corresponding public key from boot ROM 135 or flash EPROM 141 (processing block 403). As noted above in the memory maps of boot ROM 135 and flash EPROM 141, the public keys for resident application programs may be stored in boot ROM 135 or in flash EPROM 141. Alternatively, set top box 100 may be constructed so that the public keys for some resident applications are stored in boot ROM 135 and the public keys for the remaining resident applications are stored in flash EPROM 141. Next program 400 verifies the signature portion of the resident application (processing block 404). This signature verification process is the

same as previously described in conjunction with verification of RTOS 210.

[0035] Following the verification, program 400 tests the verified signature portion to determine if the resident application supports security (decision block 405). It is contemplated that any resident application that interacts with program content received from television cable 101, direct satellite receiver front end 103 or DVD 150 will require security. Other resident applications may require security at the option of the application program vendor. Program 400 bypasses other steps, load the resident application into DRAM 143 and starts the application program (processing block 410) if this signature portion indicates a non-secure use.

[0036] If the verified signature portion indicates that the resident application is to support secure applications (decision block 405), then program 400 tests to determine if the resident application can be verified as correct (decision block 406). The trap door function of the private key encryption with public key decryption makes it a very difficult task to modify the resident application program without producing an unpredictable modification of signature portion, thus enabling verification of the authorization of the resident application.

[0037] If the signature portion is not verified as secure, then program 400 indicates that the resident application is non-secure (processing block 407). Thereafter program 400 takes remedial action (processing block 408). This remedial action could be any of the many forms described above.

[0038] If the signature portion is verified as secure, then program 400 indicates that the resident application as verified (processing block 409). Thereafter program 400 starts the resident application by transferring at least part of its program code to DRAM 143 and transferring control via a jump instruction. It is contemplated that resident application programs will have access to less than all of the digital media processor functions accessible via RTOS 210.

[0039] The entire resident application could be encrypted using the private key as described above. The steps illustrated in Figure 6 would be similar except that the entire resident application must be decrypted using the public key rather than just the signature portion. As previously described, using this technique means that an unauthorized program will probably crash and disable set top box 110.

[0040] Figure 7 is a flow chart 500 of an example of verification of a downloaded program. Following the command to start downloading an application program (processing block 501), program 500 downloads the application as stores it in DRAM 143 (processing block 502). Then program 500 reads the corresponding signature portion of the downloaded application stored in DRAM 143 (processing block 503). Program 500 next reads the corresponding public key from boot ROM 135 or flash EPROM 141 (processing block 504). As noted above in the memory maps of boot ROM 135 and flash

EPROM 141, the public keys for resident application programs may be stored in either boot ROM 135 or in flash EPROM 141. Next program 500 runs signature verification on the downloaded application program (processing block 505). This signature verification process is the same as previously described in conjunction with verification of RTOS 210. A secure application program will have a signature portion that permits verification of the entire downloaded application program. A non-secure application program will have a verifiable signature stub.

[0041] Program 500 next tests to determine if the signature or signature stub has been verified (decision block 506). If the signature or signature stub has not been verified as proper, then program 500 would indicate a security violation (processing block 507) and take remedial action (processing block 508). This remedial action could be any of the many forms described above. In addition, another possible remedial action in this instance is to make an further attempt to download this application. Thus program 500 could loop back to processing block 502 to repeat the download. This remedial action would permit recovery if an authorized application was corrupted, such as by noise or the like, during download. If this option is used, it is preferable to abort this loop if after a predetermined number of signature verification failures.

[0042] Following successful verification of the signature or signature stub, program 500 tests the verified signature portion to determine if the downloaded application supports security (decision block 509). Program 500 bypasses other steps, stores and runs the downloaded application program (processing block 512), if this signature portion indicates a non-secure use. Note that the downloaded application program may be loaded into flash EPROM 141 if it is intended to be another resident application or into DRAM 143 if it is intended to be a transient application.

[0043] If the verified signature portion indicates that the downloaded application program supports secure applications (decision block 509), then program 500 tests to determine if the downloaded application can be verified as correct (decision block 511). The trap door function makes it a very difficult task to modify the downloaded application program without producing an unpredictable modification of signature portion, thus enabling verification of the authorization of the downloaded application program.

[0044] If the downloaded application program is not verified as correct (decision block 510), then program 500 indicates that the downloaded application is non-secure (processing block 507). Thereafter program 500 takes remedial action (processing block 508). This remedial action could be any of the many forms described above and may include making a further attempt to download this application program.

[0045] If the downloaded application is verified as correct (decision block 510), then program 500 indicates

the downloaded application is secure (processing block 511). Thereafter program 500 stores and runs the downloaded application program (processing block 512). As described above, this storage will be in flash EPROM 141 if the application is a resident application or in DRAM 143 if the application is a transient application. Program 500 starts the downloaded application program by transferring at least part of its program code to DRAM 143 and transferring control via a jump instruction.

[0046] The entire downloaded application program could be encrypted using the private key as described above. The steps illustrated in Figure 7 would be similar except that the entire downloaded application must be decrypted using the public key rather than just verifying the signature portion. As previously described, using this technique means that an unauthorized program will probably crash and disable set top box 110.

[0047] This security technique relies upon the security of boot ROM 135. Since boot ROM 135 is located on the sane integrated circuit as the other parts of digital media processor 130 and it is a read-only, it is not subject to unauthorized change. Therefore the verification function cannot be changed to verify a unauthorized RTOS. Many of the security functions will be available only to the RTOS based upon program privilege levels. Thus most security functions cannot be easily compromised. The private key used for encryption will only be known to the RTOS supplier, or only to the manufacturer of digital media processor 130. In addition the public key needed to verify the signature or to decrypt the RTOS is also in the boot ROM. This prevents substitution of another public key in an attempt to cause digital media processor 130 to verify an unauthorized RTOS. Additionally, the resident applications are also secure. The private keys for resident applications can be known only by the application owner, or by the service provider who authorizes the application.

[0048] The above private key/public key signature verification system will protect against most security attacks. However, if the private key used to authenticate the RTOS is compromised, the security may be defeated by replacing the RTOS with an unauthorized RTOS which will still look authentic.

[0049] The simplest way to detect a modified RTOS would be to check the resident RTOS against the authorized program. An application program, such as a diagnostic program, could read certain memory locations in the RTOS to see if they contain the expected values. This may not always reveal unauthorized substitution of another RTOS. Many complex data processors such as would be used to embody digital media processor 130 support virtual memory. In a virtual memory environment, the RTOS is quite capable of virtualising itself. Thus the unauthorized RTOS would intercept the confirming read attempts and return the results that the diagnostic application expects from a copy of the authorized RTOS. However, this unauthorized RTOS

would run instead of the original RTOS consequently compromising security. The present application propose a technique which assures that an application can access a portion of memory directly without being intercepted and translated to a virtual address by the RTOS.

[0050] Figure 8 illustrates in block diagram form a translation look-aside buffer (TLB) 137 having a locked page in accordance with the teachings of the present application. Virtual memory applications translate a virtual address into a physical address. As is known in the art, TLB 137 receives a virtual address on bus 601 and supplies a corresponding physical address on bus 602. A predetermined number of most significant address bits of the virtual address are supplied to a plurality of comparators 621, 623, 625 and 627. The remaining least significant address bits of the virtual address on bus 601 are passed unchanged to the corresponding bits of physical address on bus 620. Each comparator 621, 623, 625 and 627 has a corresponding virtual address register 611, 613, 615 and 617, respectively. The comparators 621, 623, 625 and 627 determine if the predetermined number of most significant bits of the virtual address on bus 601 matches the contents of the respective registers 611, 613, 615 and 617. Each comparator 621, 623, 625 and 627 supplied match indication to multiplexer 650. Multiplexer 650 supplies the predetermined number of most significant bits from one of the physical address registers 641, 643, 645 and 647. The physical address register selected by multiplexer 650 corresponds to the comparator 621, 623, 625 or 627 detecting a match. These most significant physical address bits selected by multiplexer 650 are supplied to the most significant bits of the physical address on bus 602. Thus TLB 137 substitutes a predetermined number of bits of a physical address for the same number of bits of the virtual address. The number of possible substitutions enabled by the virtual address register and its corresponding comparator and physical address register is limited only by considerations of operation code space to access the registers and the amount of space occupied by the TLB. In the prior art, virtual address registers 611, 613, 615 and 617 and physical address registers 641, 643, 645 and 647 are alterable via software. Thus the real time operating system has control of the mapping of virtual addresses to physical addresses.

[0051] In the preferred embodiments of the disclosed secure computing system one of the virtual address registers and the corresponding physical address register are fixed upon manufacture. In the preferred embodiment this pair of registers are mask programmable at metal layers, permitting the locked page to be selected upon manufacture of the integrated circuit including TLB 137 but unalterable following manufacture. Figure 8 illustrates a fixed virtual address register 611 and its corresponding fixed physical address register 641. In the preferred embodiment the virtual address stored in fixed virtual address register 621 equals the physical

address stored in fixed physical address register 641. In the preferred embodiment, the critical code to be protected from relocation will be stored in flash EPROM 141 within the boundary of physical addresses covered by this virtual address register. Attempts to write to either fixed virtual address register 611 or fixed physical address register 641 will fail because these registers are fixed in hardware. Preferably there will be no faults or errors generated by an attempt to modify these registers. Alternatively, neither the fixed virtual address register 611 nor the fixed physical address register 641 are accessible via the instruction set architecture. Since the reason that fixed virtual address register 611 or fixed physical address register 641 are fixed is to prevent alteration, no access via the instruction set architecture would ever be required.

[0052] A further feature of the disclosed embodiment of the present application is illustrated in Figure 8. Note that the match indication from comparator 621 is supplied directly to multiplexer 650. The match indication from other comparators form the noninverting input to respective AND gates 633, 635 and 637. Each of these AND gates 633, 635 and 637 receives the match indication from comparator 621 on an inverting input. Thus a match indication from comparator 621 prevents supply of a match indication to multiplexer 650 from any other comparator. This prevents an unauthorized person from leaving the original RTOS in place to respond to security queries while attempting to run an unauthorized RTOS from a relocated part of memory. Any memory accesses to the physical memory addresses of virtual address register 611 and physical address register 641 cannot be relocated but are directed to the physical address of the original RTOS.

[0053] With the disclosed embodiment of the present application an unauthorized attempt to relocate the RTOS may occur, but no actual address translation will take place. Thus if the original RTOS is always located in this memory area, a diagnostic program can read signature locations with assurance that the original physical locations are being accessed. Thus the diagnostic program can determine if the RTOS is compromised, and take appropriate remedial action. This remedial action may include any of the remedial actions previously described.

[0054] The set top box 100 illustrated in Figure 1 includes an additional potential security problem. DRAM 143 stores a video data stream that has been decrypted but not decompressed. This video data is stored in compressed video FIFO buffer 280. It is possible for an unauthorized person to intercept this data as it is being transferred from digital media processor 130 to DRAM 143 or as it is being transferred from DRAM 143 and digital media processor 130. These data transfers will be interleaved with other data traffic between digital media processor 130 and DRAM 143, but it is feasible to separate the compressed video data. Because the video is compressed, a minimal amount of

memory would be required to store this data. Some content providers would like to prevent their video programming from such interception. Note that interception of the video data stream at this point would permit generation of plural, identical and immediately viewable copies of the video.

[0055] Figure 9 illustrates in flow chart form a process preventing such unauthorized interception. Following reception of the video data stream (processing block 701) digital media processor 130 decrypts the video data stream (processing block 702). This decryption is subject to security procedures to ensure that the user is authorized to view this video data stream. Following this decryption of the source program, digital media processor encrypts the video data stream again (processing block 703). In this instance a relatively simple encryption is used, such as a simplified DES algorithm. The encryption key is preferably derived from the chip identity number stored in chip identity register 133. This encrypted data is stored in compressed video FIFO buffer 280 (processing block 704). At the appropriate time, the video data is recalled from compressed video FIFO buffer 280 (processing block 705). The recalled data is decrypted using the encryption key derived from the chip identity number (706). This data is then ready for further processing (processing block 707).

[0056] This technique has the advantage that the compressed video data stream temporarily stored in compressed video FIFO buffer 280 can only be read by the particular digital media processor 130. The chip identity number is unique to that particular digital media processor. The video data cannot be viewed by any other means, even another identical set top box system 100 without breaking the code. This is believed adequate security by most content providers. Additionally, the encryption and decryption is transparent to the user. There only needs to be a small additional processing capacity available within digital media processor 130 beyond the minimal requirement of the particular application.

[0057] Another potential security problem is created by the hardware debugger/ emulator. The semiconductor manufacturer of digital media processor 130 will generally also sell hardware debugger/emulator systems to application program developers, including operating system developers. Generally such hardware debugger/emulator systems by design have unlimited access to all of memory, including "private" areas. Thus a hardware debugger/emulator system of the type known in the art would permit unauthorized breach of the security of set top box system 100.

[0058] The following modification to the hardware debugger/emulator system will guard against this potential security problem. The hardware debugger/emulator will operate in two modes, a process mode and a raw mode. In the process mode, the hardware debugger/emulator may only access a particular process or application program. All system access is permitted

in the raw mode.

[0059] Figure 10 is a flow chart illustrating the process of selecting the mode at the hardware debugger/emulator. Upon start of the hardware debugger/ emulator (processing block 801), process 800 reads the signature portion 211 of RTOS 210 stored in flash EPROM 141 (processing block 802). Process 800 next reads the RTOS public key 203 from boot EPROM 135 (processing block 803). Next process 800 verifies the signature portion 211 of RTOS 210 (processing block 804). Following the verification, process 800 tests the verified signature portion to determine if RTOS 210 supports secure applications (decision block 805). As previously described, digital media processor 130 could be embodied in applications not requiring the security of set top boxes. In such applications, the verified signature portion 211 indicates that the RTOS need not be secured. If this is the case, then process 800 bypasses other steps activates the hardware debugger/emulator in raw mode (processing block 806).

[0060] If the RTOS supports secure applications (decision block 805), then process 800 checks to determine if the chip identity number stored in chip identity register 133 is of the subset of possible chip identity numbers that permit the raw mode for secure applications (decision block 807). Some program developers, particularly RTOS developers, will need access to the raw mode of the hardware debugger/emulator. The present application contemplates that a bit or bits or some subset of the possible coding of the chip identity number will be reserved for hardware debugger/emulators supporting this use. Thus only a certain limited number of the digital media processors 130 will permit raw mode operation of the hardware debugger/emulator in environments supporting the security described above. The manufacturer of digital media processor 130 will supply these particularly identified chips only to trusted program developers.

[0061] If the chip identity number does not permit raw mode operation (decision block 807), process 800 reads a token from the particular process or application program under development in the hardware debugger/emulator. Process 800 then determines if this token is verified as proper (decision block 809). This process could take place using the private key encryption and public key decryption described above, or another verification procedure could be employed. If the token is not verified (decision block 809), then process 800 take appropriate remedial action (processing block 810). The various types of remedial action that could be taken have already been described. If the token is verified (decision block 809), then process 800 activates the hardware debugger/emulator in process mode (processing block 811). In the process mode, the hardware debugger/emulator may only access a particular process or application program corresponding to the verified token.

[0062] This process satisfies all the requirements of

the users. Program developers who use digital media processor 130 is non-secure application will have complete access to the functions of the hardware debugger/emulator. Program developers who use digital media processor 130 is secure applications will have access limited, Most of those program developers will use the secure RTOS and have access only to their own programs as identified by the token encrypted with their corresponding private key. RTOS developers will have complete system access but only to particular digital media processors having the proper chip identity numbers. Thus the manufacturer of digital media processor 130 can have the proper level of control in order protect the security of set top box systems 100.

[0063] The exemplary embodiments of this patent application have been described in conjunction with a particular type system requiring computer security, i.e. the set top box. Those skilled in the art would realize that the use of these security techniques are not limited to this example. Particularly, almost any computer system requiring that some functions have a degree of security may employ these techniques.

Claims

1. A secure computing system comprising:

- a non-volatile memory for storing program code for at least one program, said program code including a verification code encrypted with a predetermined private key;
- a data processor for data manipulation under program control disposed on an integrated circuit, said data processor executing a program stored at a predetermined address upon each initial application of electric power;
- a read only memory disposed on said integrated circuit for storing a public key corresponding to said predetermined private key and for storing an initialization program stored beginning at said predetermined address, said initialization program including instructions for causing said data processor to employ said public key to decrypt said verification code of said at least one program stored in said non-volatile memory, said initialization program further including instructions for causing said data processor to indicate verification of security of said program or non-verification of security of said program.

2. The secure computing system of claim 1, wherein said at least one program stored in said non-volatile memory comprises a real time operating system for said secure computing system.

3. The secure computing system of claim 2, wherein said at least one program stored in said non-volatile

5 memory includes an application program for cooperating with said real time operating system, said application program including a second verification code encrypted with a predetermined second private key;

said read only memory further being arranged for storing a second public key corresponding to said predetermined second private key, and said initialization program further including instructions for causing said data processor to employ said second public key to decrypt said second verification code of said application program stored in said non-volatile memory, and to indicate verification of security of said application program or non-verification of security of said application program.

4. The secure computing system of claim 1, wherein 20 said at least one program stored in said non-volatile memory includes a real time operating system for said secure computing system and a plurality of application programs for cooperating with said real time operating system, each of said application programs including a corresponding verification code encrypted with a predetermined private key;

25 said read only memory being arranged for storing a public key corresponding to each of said predetermined private keys, and said initialization program further including instructions for causing said data processor to employ said corresponding public key to decrypt said verification code of each of said plurality of application programs stored in said non-volatile memory, and to indicate verification of security of each of said plurality of application programs or non-verification of security of each of said application programs.

30 40 45 50 55 60 65 70 75 80 85 90 95 100 105 110 115 120 125 130 135 140 145 150 155 160 165 170 175 180 185 190 195 200 205 210 215 220 225 230 235 240 245 250 255 260 265 270 275 280 285 290 295 300 305 310 315 320 325 330 335 340 345 350 355 360 365 370 375 380 385 390 395 400 405 410 415 420 425 430 435 440 445 450 455 460 465 470 475 480 485 490 495 500 505 510 515 520 525 530 535 540 545 550 555 560 565 570 575 580 585 590 595 600 605 610 615 620 625 630 635 640 645 650 655 660 665 670 675 680 685 690 695 700 705 710 715 720 725 730 735 740 745 750 755 760 765 770 775 780 785 790 795 800 805 810 815 820 825 830 835 840 845 850 855 860 865 870 875 880 885 890 895 900 905 910 915 920 925 930 935 940 945 950 955 960 965 970 975 980 985 990 995 1000 1005 1010 1015 1020 1025 1030 1035 1040 1045 1050 1055 1060 1065 1070 1075 1080 1085 1090 1095 1100 1105 1110 1115 1120 1125 1130 1135 1140 1145 1150 1155 1160 1165 1170 1175 1180 1185 1190 1195 1200 1205 1210 1215 1220 1225 1230 1235 1240 1245 1250 1255 1260 1265 1270 1275 1280 1285 1290 1295 1300 1305 1310 1315 1320 1325 1330 1335 1340 1345 1350 1355 1360 1365 1370 1375 1380 1385 1390 1395 1400 1405 1410 1415 1420 1425 1430 1435 1440 1445 1450 1455 1460 1465 1470 1475 1480 1485 1490 1495 1500 1505 1510 1515 1520 1525 1530 1535 1540 1545 1550 1555 1560 1565 1570 1575 1580 1585 1590 1595 1600 1605 1610 1615 1620 1625 1630 1635 1640 1645 1650 1655 1660 1665 1670 1675 1680 1685 1690 1695 1700 1705 1710 1715 1720 1725 1730 1735 1740 1745 1750 1755 1760 1765 1770 1775 1780 1785 1790 1795 1800 1805 1810 1815 1820 1825 1830 1835 1840 1845 1850 1855 1860 1865 1870 1875 1880 1885 1890 1895 1900 1905 1910 1915 1920 1925 1930 1935 1940 1945 1950 1955 1960 1965 1970 1975 1980 1985 1990 1995 2000 2005 2010 2015 2020 2025 2030 2035 2040 2045 2050 2055 2060 2065 2070 2075 2080 2085 2090 2095 2100 2105 2110 2115 2120 2125 2130 2135 2140 2145 2150 2155 2160 2165 2170 2175 2180 2185 2190 2195 2200 2205 2210 2215 2220 2225 2230 2235 2240 2245 2250 2255 2260 2265 2270 2275 2280 2285 2290 2295 2300 2305 2310 2315 2320 2325 2330 2335 2340 2345 2350 2355 2360 2365 2370 2375 2380 2385 2390 2395 2400 2405 2410 2415 2420 2425 2430 2435 2440 2445 2450 2455 2460 2465 2470 2475 2480 2485 2490 2495 2500 2505 2510 2515 2520 2525 2530 2535 2540 2545 2550 2555 2560 2565 2570 2575 2580 2585 2590 2595 2600 2605 2610 2615 2620 2625 2630 2635 2640 2645 2650 2655 2660 2665 2670 2675 2680 2685 2690 2695 2700 2705 2710 2715 2720 2725 2730 2735 2740 2745 2750 2755 2760 2765 2770 2775 2780 2785 2790 2795 2800 2805 2810 2815 2820 2825 2830 2835 2840 2845 2850 2855 2860 2865 2870 2875 2880 2885 2890 2895 2900 2905 2910 2915 2920 2925 2930 2935 2940 2945 2950 2955 2960 2965 2970 2975 2980 2985 2990 2995 3000 3005 3010 3015 3020 3025 3030 3035 3040 3045 3050 3055 3060 3065 3070 3075 3080 3085 3090 3095 3100 3105 3110 3115 3120 3125 3130 3135 3140 3145 3150 3155 3160 3165 3170 3175 3180 3185 3190 3195 3200 3205 3210 3215 3220 3225 3230 3235 3240 3245 3250 3255 3260 3265 3270 3275 3280 3285 3290 3295 3300 3305 3310 3315 3320 3325 3330 3335 3340 3345 3350 3355 3360 3365 3370 3375 3380 3385 3390 3395 3400 3405 3410 3415 3420 3425 3430 3435 3440 3445 3450 3455 3460 3465 3470 3475 3480 3485 3490 3495 3500 3505 3510 3515 3520 3525 3530 3535 3540 3545 3550 3555 3560 3565 3570 3575 3580 3585 3590 3595 3600 3605 3610 3615 3620 3625 3630 3635 3640 3645 3650 3655 3660 3665 3670 3675 3680 3685 3690 3695 3700 3705 3710 3715 3720 3725 3730 3735 3740 3745 3750 3755 3760 3765 3770 3775 3780 3785 3790 3795 3800 3805 3810 3815 3820 3825 3830 3835 3840 3845 3850 3855 3860 3865 3870 3875 3880 3885 3890 3895 3900 3905 3910 3915 3920 3925 3930 3935 3940 3945 3950 3955 3960 3965 3970 3975 3980 3985 3990 3995 4000 4005 4010 4015 4020 4025 4030 4035 4040 4045 4050 4055 4060 4065 4070 4075 4080 4085 4090 4095 4100 4105 4110 4115 4120 4125 4130 4135 4140 4145 4150 4155 4160 4165 4170 4175 4180 4185 4190 4195 4200 4205 4210 4215 4220 4225 4230 4235 4240 4245 4250 4255 4260 4265 4270 4275 4280 4285 4290 4295 4300 4305 4310 4315 4320 4325 4330 4335 4340 4345 4350 4355 4360 4365 4370 4375 4380 4385 4390 4395 4400 4405 4410 4415 4420 4425 4430 4435 4440 4445 4450 4455 4460 4465 4470 4475 4480 4485 4490 4495 4500 4505 4510 4515 4520 4525 4530 4535 4540 4545 4550 4555 4560 4565 4570 4575 4580 4585 4590 4595 4600 4605 4610 4615 4620 4625 4630 4635 4640 4645 4650 4655 4660 4665 4670 4675 4680 4685 4690 4695 4700 4705 4710 4715 4720 4725 4730 4735 4740 4745 4750 4755 4760 4765 4770 4775 4780 4785 4790 4795 4800 4805 4810 4815 4820 4825 4830 4835 4840 4845 4850 4855 4860 4865 4870 4875 4880 4885 4890 4895 4900 4905 4910 4915 4920 4925 4930 4935 4940 4945 4950 4955 4960 4965 4970 4975 4980 4985 4990 4995 5000 5005 5010 5015 5020 5025 5030 5035 5040 5045 5050 5055 5060 5065 5070 5075 5080 5085 5090 5095 5100 5105 5110 5115 5120 5125 5130 5135 5140 5145 5150 5155 5160 5165 5170 5175 5180 5185 5190 5195 5200 5205 5210 5215 5220 5225 5230 5235 5240 5245 5250 5255 5260 5265 5270 5275 5280 5285 5290 5295 5300 5305 5310 5315 5320 5325 5330 5335 5340 5345 5350 5355 5360 5365 5370 5375 5380 5385 5390 5395 5400 5405 5410 5415 5420 5425 5430 5435 5440 5445 5450 5455 5460 5465 5470 5475 5480 5485 5490 5495 5500 5505 5510 5515 5520 5525 5530 5535 5540 5545 5550 5555 5560 5565 5570 5575 5580 5585 5590 5595 5600 5605 5610 5615 5620 5625 5630 5635 5640 5645 5650 5655 5660 5665 5670 5675 5680 5685 5690 5695 5700 5705 5710 5715 5720 5725 5730 5735 5740 5745 5750 5755 5760 5765 5770 5775 5780 5785 5790 5795 5800 5805 5810 5815 5820 5825 5830 5835 5840 5845 5850 5855 5860 5865 5870 5875 5880 5885 5890 5895 5900 5905 5910 5915 5920 5925 5930 5935 5940 5945 5950 5955 5960 5965 5970 5975 5980 5985 5990 5995 6000 6005 6010 6015 6020 6025 6030 6035 6040 6045 6050 6055 6060 6065 6070 6075 6080 6085 6090 6095 6100 6105 6110 6115 6120 6125 6130 6135 6140 6145 6150 6155 6160 6165 6170 6175 6180 6185 6190 6195 6200 6205 6210 6215 6220 6225 6230 6235 6240 6245 6250 6255 6260 6265 6270 6275 6280 6285 6290 6295 6300 6305 6310 6315 6320 6325 6330 6335 6340 6345 6350 6355 6360 6365 6370 6375 6380 6385 6390 6395 6400 6405 6410 6415 6420 6425 6430 6435 6440 6445 6450 6455 6460 6465 6470 6475 6480 6485 6490 6495 6500 6505 6510 6515 6520 6525 6530 6535 6540 6545 6550 6555 6560 6565 6570 6575 6580 6585 6590 6595 6600 6605 6610 6615 6620 6625 6630 6635 6640 6645 6650 6655 6660 6665 6670 6675 6680 6685 6690 6695 6700 6705 6710 6715 6720 6725 6730 6735 6740 6745 6750 6755 6760 6765 6770 6775 6780 6785 6790 6795 6800 6805 6810 6815 6820 6825 6830 6835 6840 6845 6850 6855 6860 6865 6870 6875 6880 6885 6890 6895 6900 6905 6910 6915 6920 6925 6930 6935 6940 6945 6950 6955 6960 6965 6970 6975 6980 6985 6990 6995 7000 7005 7010 7015 7020 7025 7030 7035 7040 7045 7050 7055 7060 7065 7070 7075 7080 7085 7090 7095 7100 7105 7110 7115 7120 7125 7130 7135 7140 7145 7150 7155 7160 7165 7170 7175 7180 7185 7190 7195 7200 7205 7210 7215 7220 7225 7230 7235 7240 7245 7250 7255 7260 7265 7270 7275 7280 7285 7290 7295 7300 7305 7310 7315 7320 7325 7330 7335 7340 7345 7350 7355 7360 7365 7370 7375 7380 7385 7390 7395 7400 7405 7410 7415 7420 7425 7430 7435 7440 7445 7450 7455 7460 7465 7470 7475 7480 7485 7490 7495 7500 7505 7510 7515 7520 7525 7530 7535 7540 7545 7550 7555 7560 7565 7570 7575 7580 7585 7590 7595 7600 7605 7610 7615 7620 7625 7630 7635 7640 7645 7650 7655 7660 7665 7670 7675 7680 7685 7690 7695 7700 7705 7710 7715 7720 7725 7730 7735 7740 7745 7750 7755 7760 7765 7770 7775 7780 7785 7790 7795 7800 7805 7810 7815 7820 7825 7830 7835 7840 7845 7850 7855 7860 7865 7870 7875 7880 7885 7890 7895 7900 7905 7910 7915 7920 7925 7930 7935 7940 7945 7950 7955 7960 7965 7970 7975 7980 7985 7990 7995 8000 8005 8010 8015 8020 8025 8030 8035 8040 8045 8050 8055 8060 8065 8070 8075 8080 8085 8090 8095 8100 8105 8110 8115 8120 8125 8130 8135 8140 8145 8150 8155 8160 8165 8170 8175 8180 8185 8190 8195 8200 8205 8210 8215 8220 8225 8230 8235 8240 8245 8250 8255 8260 8265 8270 8275 8280 8285 8290 8295 8300 8305 8310 8315 8320 8325 8330 8335 8340 8345 8350 8355 8360 8365 8370 8375 8380 8385 8390 8395 8400 8405 8410 8415 8420 8425 8430 8435 8440 8445 8450 8455 8460 8465 8470 8475 8480 8485 8490 8495 8500 8505 8510 8515 8520 8525 8530 8535 8540 8545 8550 8555 8560 8565 8570 8575 8580 8585 8590 8595 8600 8605 8610 8615 8620 8625 8630 8635 8640 8645 8650 8655 8660 8665 8670 8675 8680 8685 8690 8695 8700 8705 8710 8715 8720 8725 8730 8735 8740 8745 8750 8755 8760 8765 8770 8775 8780 8785 8790 8795 8800 8805 8810 8815 8820 8825 8830 8835 8840 8845 8850 8855 8860 8865 8870 8875 8880 8885 8890 8895 8900 8905 8910 8915 8920 8925 8930 8935 8940 8945 8950 8955 8960 8965 8970 8975 8980 8985 8990 8995 9000 9005 9010 9015 9020 9025 9030 9035 9040 9045 9050 9055 9060 9065 9070 9075 9080 9085 9090 9095 9100 9105 9110 9115 9120 9125 9130 9135 9140 9145 9150 9155 9160 9165 9170 9175 9180 9185 9190 9195 9200 9205 9210 9215 9220 9225 9230 9235 9240 9245 9250 9255 9260 9265 9270 9275 9280 9285 9290 9295 9300 9305 9310 9315 9320 9325 9330 9335 9340 9345 9350 9355 9360 9365 9370 9375 9380 9385 9390 9395 9400 9405 9410 9415 9420 9425 9430 9435 9440 9445 9450 9455 9460 9465 9470 9475 9480 9485 9490 9495 9500 9505 9510 9515 9520 9525 9530 9535 9540 9545 9550 9555 9560 9565 9570 9575 9580 9585 9590 9595 9600 9605 9610 9615 9620 9625 9630 9635 9640 9645 9650 9655 9660 9665 9670 9675 9680 9685 9690 9695 9700 9705 9710 9715 9720 9725 9730 9735 9740 9745 9750 9755 9760 9765 9770 9775 9780 9785 9790 9795 9800 9805 9810 9815 9820 9825 9830 9835 9840 9845 9850 9855 9860 9865 9870 9875 9880 9885 9890 9895 9900 9905 9910 9915 9920 9925 9930 9935 9940 9945 9950 9955 9960 9965 9970 9975 9980 9985 9990 9995 10000 10005 10010 10015 10020 10025 10030 10035 10040 10045 10050 10055 10060 10065 10070 10075 10080 10085 10090 10095 10100 10105 10110 10115 10120 10125 10130 10135 10140 10145 10150 10155 10160 10165 10170 10175 10180 10185 10190 10195 10200 10205 10210 10215 10220 10225 10230 10235 10240 10245 10250 10255 10260 10265 10270 10275 10280 10285 10290 10295 10300 10305 10310 10315 10320 10325 10330 10335 10340 10345 10350 10355 10360 10365 10370 10375 10380 10

number of bits less than said first predetermined number of bits,

a plurality of writable virtual address registers of said second predetermined number of bits,
5
a first comparator connected to said address generator and said fixed virtual address register for comparing the contents of said first fixed address register with said second predetermined number of bits of said virtual address and indicating a match,
10
a plurality of second comparators, each connected to a corresponding virtual address register and said address generator, each for comparing the contents of said corresponding virtual address register with said second predetermined number of bits of said virtual address and indicating a match,
15
a fixed physical address register of said second predetermined number of bits,
a plurality of writable physical address registers of said second predetermined number of bits, and
20
a multiplexer connected to said memory, said address generator, said first comparator, each of said second comparators, said fixed physical address register and each of said plurality of writable physical address registers, said multiplexer responsive to a match by one of said comparators to substitute contents of a physical register corresponding to said matching comparator for most significant bits of said virtual address and thereby form a physical address supplied to said memory for memory access.

7. The secure computing system of claim 6, wherein:
40
said multiplexer is responsive to an indication of a match by said first comparator to substitute the contents of said fixed physical register for most significant bits of said virtual address.
45

8. The secure computing system of claim 6 or claim 7, wherein said fixed virtual address register and said fixed physical address register are mask programmable in manufacture.
50

9. The secure computing system of any of claims 6 to 8, wherein said plurality of writable virtual address registers and said plurality of writable physical address registers are writable upon execution of an instruction by said secure computing system;
55
said fixed virtual address register and said

fixed physical address register writable upon execution of an instruction by said secure computing system, an attempt to write to either said fixed virtual address register or said fixed physical address register via said instruction being arranged to fail to alter contents of said register and to generate no error message or fault.

10. A secure computing system comprising:
a data processor disposed on a single integrated circuit, said data processor including a chip identity read only register for storing a unique chip identity number;
a memory bi-directionally coupled to said data processor for storing data;
said data processor being programmed to:
(i) encrypt data employing at least a part of said chip identity number as an encryption key,
(ii) store said encrypted data in said memory,
(iii) recall said stored data from said memory, and
(iv) decrypt said recalled data employing at least a part of said chip identity number as decryption key.

30 11. The secure computing system of claim 10, wherein:
said data comprises a stream of video data.

12. A method of secure computing comprising the steps of:
35
encrypting a verification token for a program with private key;
storing a public key corresponding to said private key;
upon each initialization of a debugger/emulator for a secure computing system determining if said program is secure program or a non-secure program,
if said program is a non-secure program selecting a first operating mode in said debugger/emulator permitting access to said program while prohibiting access to at least one security feature of the secure computing system, and
if said program is a secure program selecting a second operating mode in said debugger/emulator permitting access to all features of the secure computing system.

55 13. The method of secure computing of claim 12, further comprising the steps of:
storing a unique chip identity number on a data

processor within the secure computing system;
if said program is a secure program testing to
determine if said unique chip identity number of
said data processor is within a predetermined
subset of possible chip identity numbers; 5
if said unique chip identity number of said data
processor is within said predetermined subset
of possible chip identity numbers selecting said
second operating mode in said debugger/emulator;
and 10
if said unique chip identity number of said data
processor is not within said predetermined
subset of possible chip identity numbers select-
ing said first operating mode in said debug-
ger/emulator. 15

14. The method of secure computing of claim 12, fur-
ther comprising the steps of:
wherein said program is an operating system for a
data processor of the secure computing system; 20

encrypting with a second private key at least a
verification token of an application program;
storing a second public key corresponding to
said second private key; 25
decrypting said application program employing
said public key as a decryption key;
indicating verification or non-verification of
security of said decrypted application program;
selecting said first operating mode in said 30
debugger/emulator if said decrypted applica-
tion program is verified as secure.

35

40

45

50

55

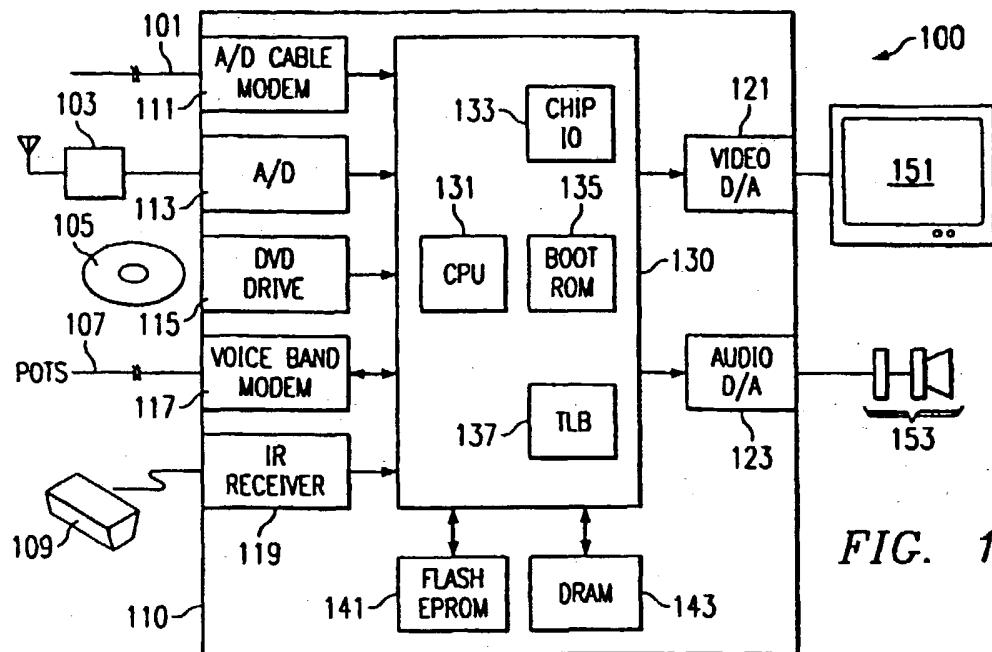


FIG. 1

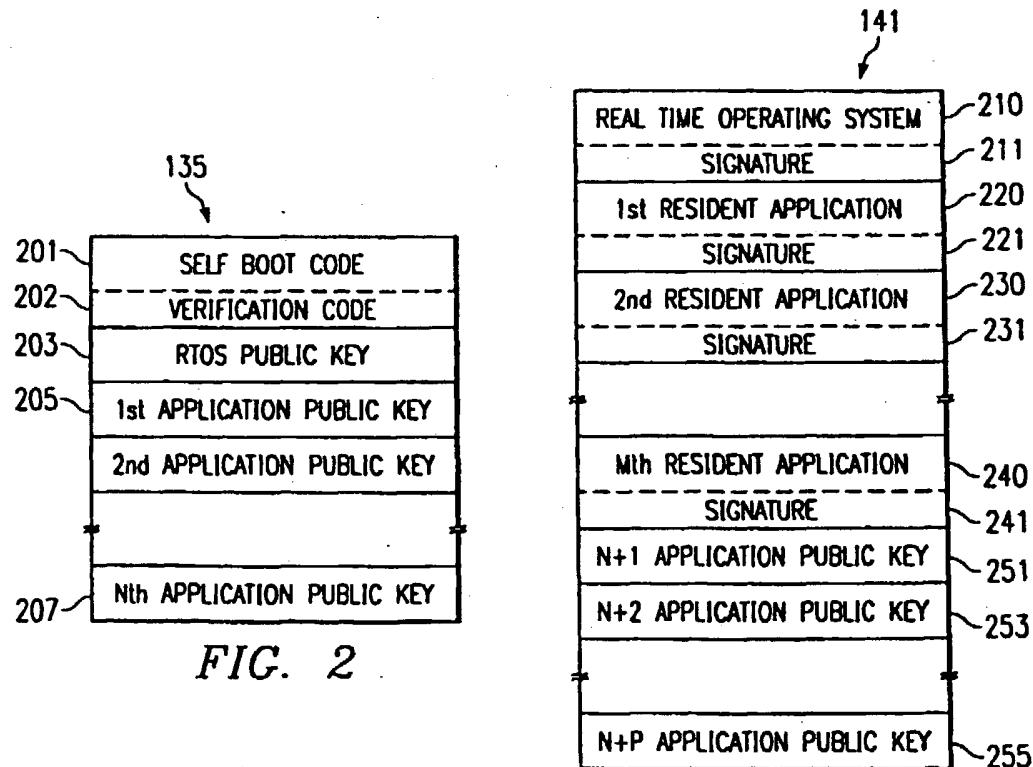


FIG. 2

FIG. 3

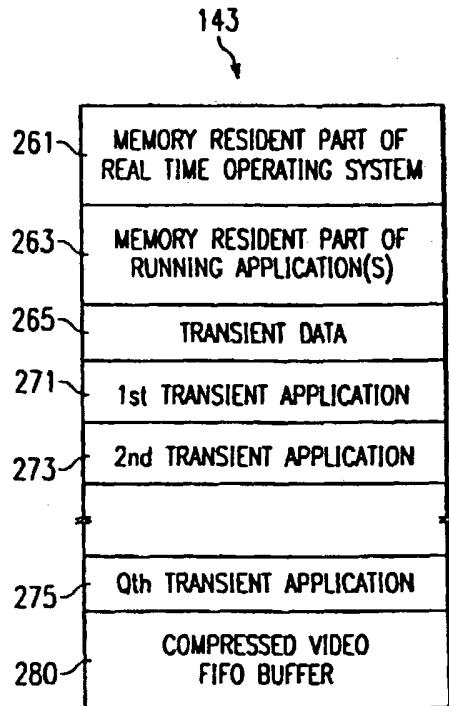


FIG. 4

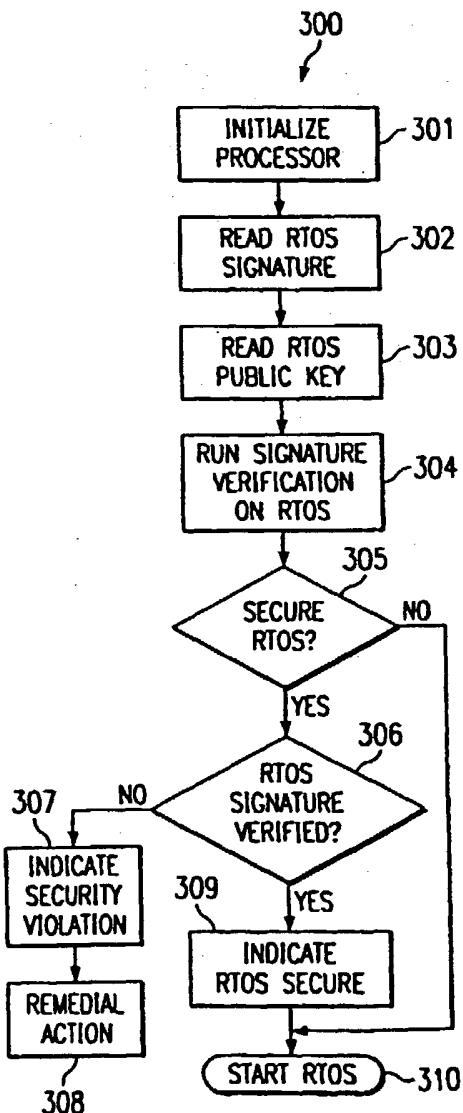
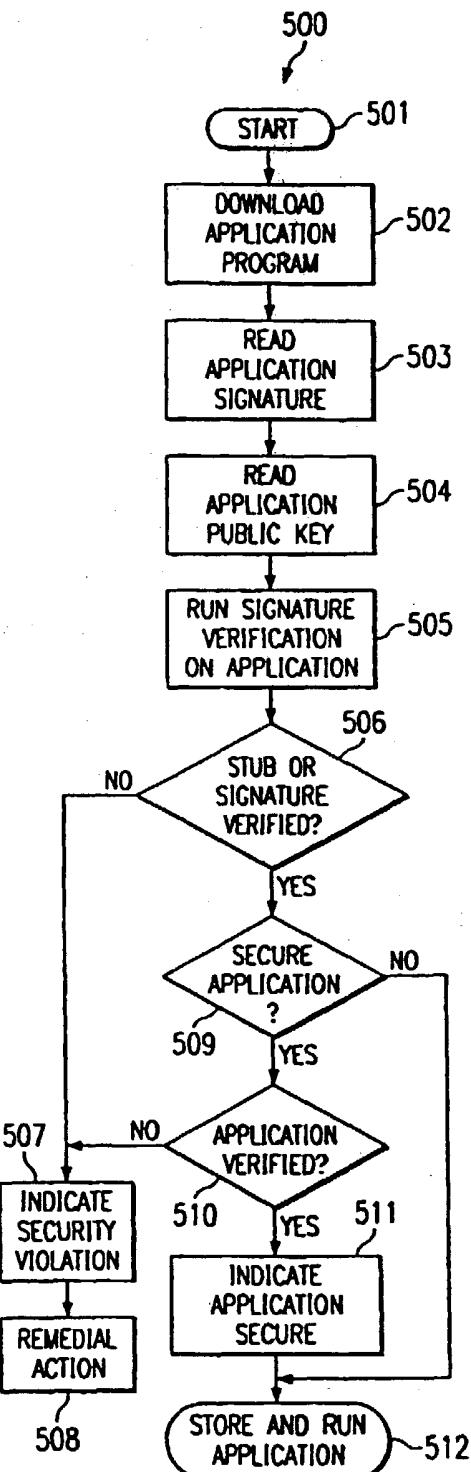
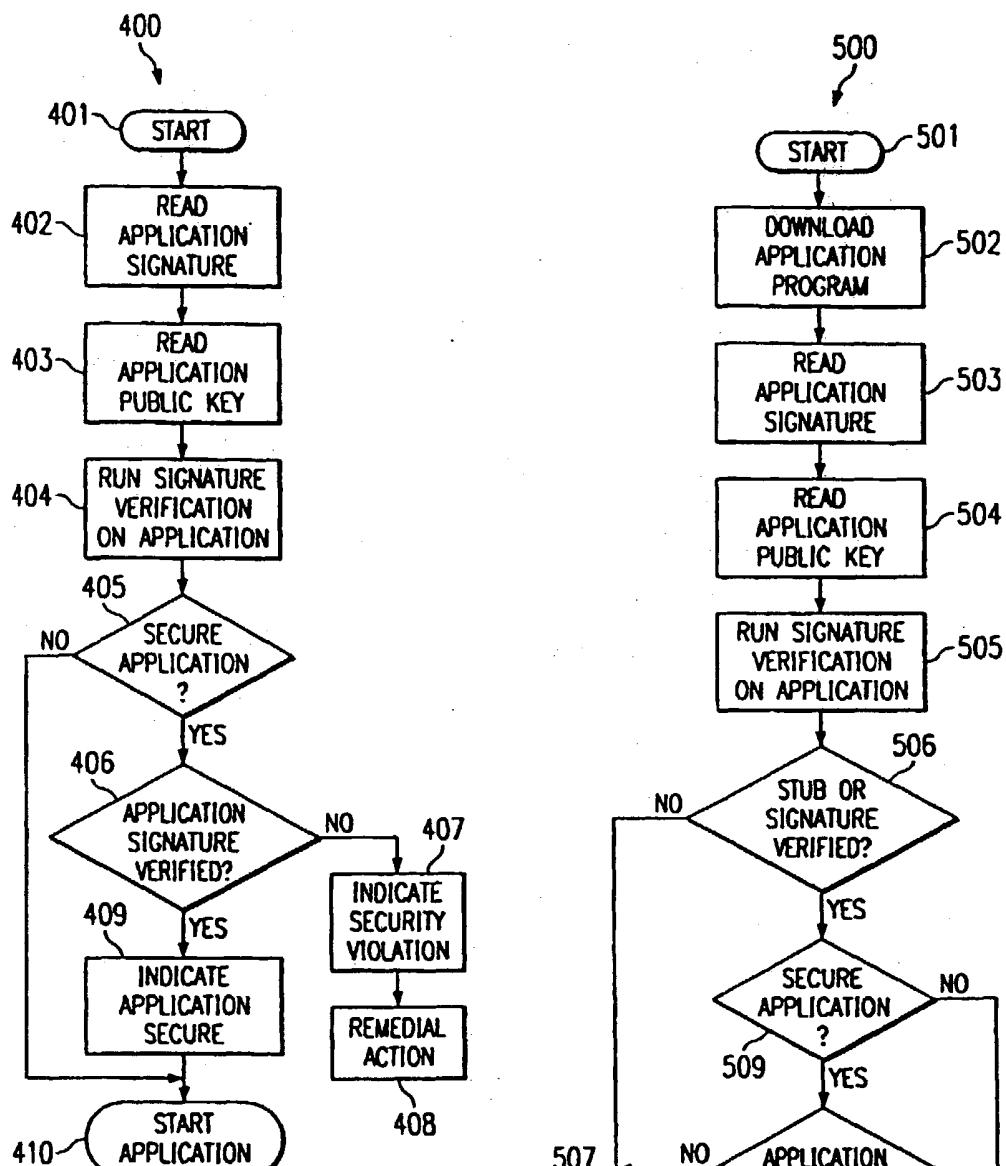


FIG. 5



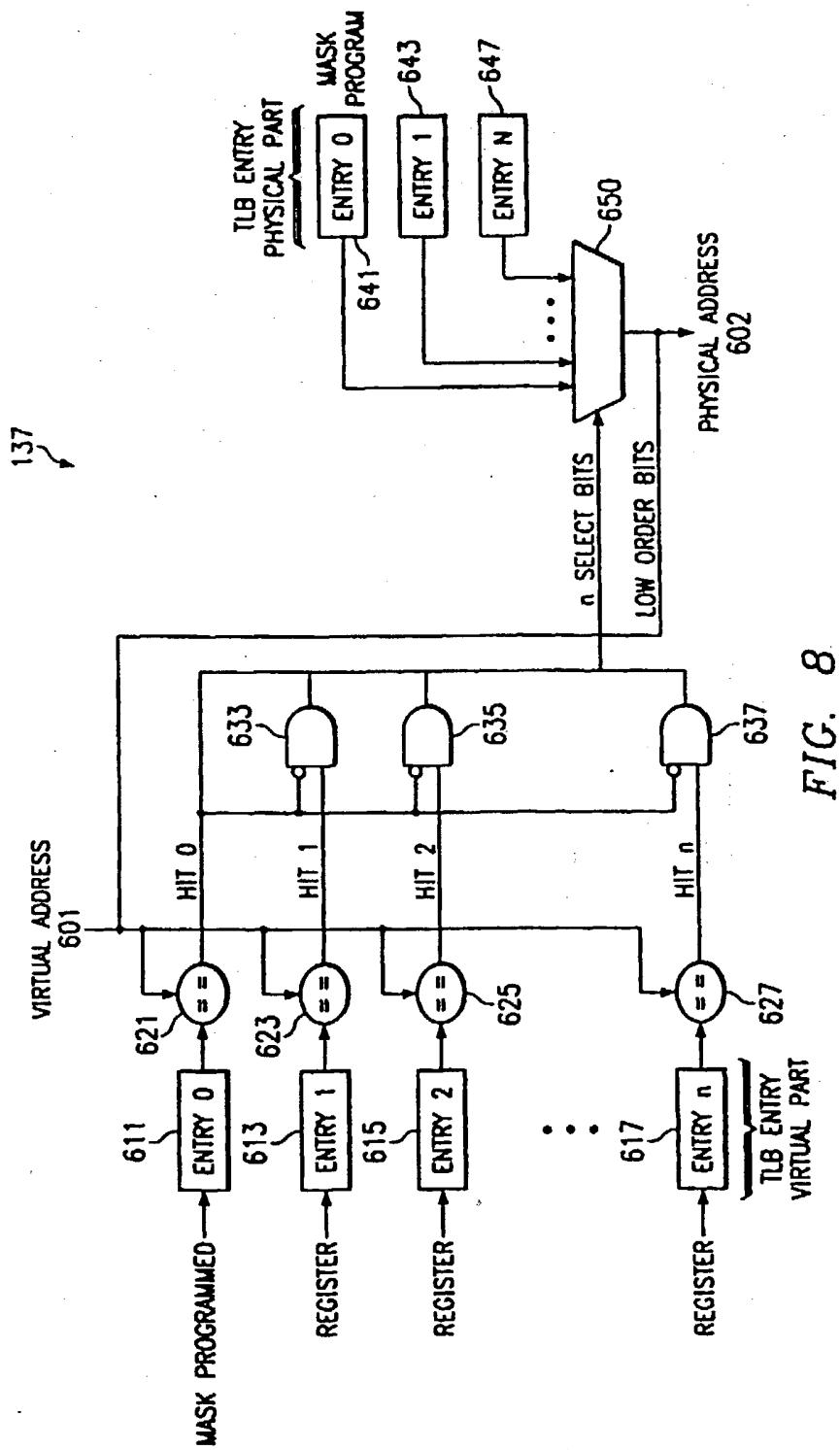


FIG. 8

